# BAB V IMPLEMENTASI

Dalam bab ini akan dijelaskan mengenai implementasi dalam program Point of Sales UD Bintang Jaya ini Implementasi sistem berisi code-code penting yang terdapat dalam Point of Sales ini. Code penting tersebut merupakan code dari fitur-fitur penting dalam sistem. Berikut penjelasan dari setiap codenya.

## 5.1 Edit User

Pada bagian ini pemilik toko atau akuntan dapat melakukan edit password dan juga membuat akun untuk para pegawainya, disini pemilik toko dapat menambahkan Kasir dan akuntan lainnya, dan juga bisa mengganti Peran ( Akuntan atau Kasir ) dan password dari user tersebut.

**Segmen Program 5.1 Edit User**

1. private void btnEdit\_Click(object sender, EventArgs e){
2. DialogResult dialogResult;
3. dialogResult = MessageBox.Show("Apakah Anda yakin ingin simpan hasil ubahan ?", "Warning", MessageBoxButtons.YesNo, MessageBoxIcon.Information);
4. if (dialogResult == DialogResult.Yes){
5. if (txtUsername.Text != "" && txtPasswordUser.Text != "" && txtNama.Text != "" && cmbRoles.SelectedIndex > -1){
6. if (txtUsername.Text.Length > 20){
7. MessageBox.Show("Username tidak boleh lebih dari 20 karakter");
8. return; }
9. if (txtPasswordUser.Text.Length > 10){
10. MessageBox.Show("Password tidak boleh lebih dari 10 karakter");
11. return;}
12. if (txtUsername.Text.Length > 50){
13. MessageBox.Show("Nama tidak boleh lebih dari 50 karakter");
14. return;}
15. try{
16. string queryUpdUser = "update userpos set user\_password = '" + txtPasswordUser.Text + "', user\_nama = '" + txtNama.Text + "', user\_role = '" + arrayRoles[cmbRoles.SelectedIndex] + "' " +
17. "where user\_username = '" + txtUsername.Text + "'";
18. if (conn.State == ConnectionState.Closed){
19. conn = new OracleConnection(connStr);
20. conn.Open();}
21. OracleCommand oracleCommand = new OracleCommand(queryUpdUser, conn);
22. oracleCommand.ExecuteNonQuery();
23. MessageBox.Show("Berhasil update data username " + txtUsername.Text + "!");
24. resetAll();}
25. catch (Exception error){
26. MessageBox.Show("Error saat update data User \n" + error.Message + "\nHubungi Developer segera !");
27. } }

Pada segmen program 5.1, baris 5 - 14 terjadi pengecekan terhadap field yang diisi agar panjang tidak melebihi batas. Pada baris 16 query untuk update disusun. Pada baris 21-22 query dieksekusi. Apabila eksekusi berhasil maka akan muncul *messagebox* bahwa berhasil mengubah data user. Apabila eksekusi query gagal akan masuk ke baris 25-27 dan muncul *messagebox* bahwa terjadi kesalahan saat menyimpan data user.

## 5.2 Tambah User

Pada bagian ini Pemilik toko atau akuntan dapat menambahkan user, yang dimana terdapat dua role yang dapat ditambahkan yaitu role Akuntan dan juga role Kasir, hanya role akuntan saja yang bisa melihat, mengganti, dan mengubah role dari Kasir.

**Segmen Program 5.2 Tambah User**

1. private void btnTambah\_Click(object sender, EventArgs e){
2. DialogResult dialogResult;
3. dialogResult = MessageBox.Show("Apakah Anda yakin ingin simpan hasil ubahan password?", "Warning", MessageBoxButtons.YesNo, MessageBoxIcon.Information);
4. if (dialogResult == DialogResult.Yes){
5. if (txtUsername.Text != "" && txtPasswordUser.Text != "" && txtNama.Text != "" && cmbRoles.SelectedIndex > -1){
6. if (txtUsername.Text.Length > 20)
7. {
8. MessageBox.Show("Username tidak boleh lebih dari 20 karakter");
9. return;}
10. if (txtPasswordUser.Text.Length > 10){
11. MessageBox.Show("Password tidak boleh lebih dari 10 karakter");
12. return;}
13. if (txtUsername.Text.Length > 50){
14. MessageBox.Show("Nama tidak boleh lebih dari 50 karakter");
15. return;}
16. try{
17. string queryInsUser = "insert into userpos values ('" + txtUsername.Text + "', '" + txtPasswordUser.Text + "', '" + txtNama.Text + "', '" + arrayRoles[cmbRoles.SelectedIndex] + "', 0)";
18. if (conn.State == ConnectionState.Closed){
19. conn = new OracleConnection(connStr);
20. conn.Open();}
21. OracleCommand oracleCommand = new OracleCommand(queryInsUser, conn);
22. oracleCommand.ExecuteNonQuery();
23. MessageBox.Show("Berhasil simpan data username " + txtUsername.Text + " !");
24. resetAll();}
25. catch (Exception error)
26. {
27. MessageBox.Show("Error saat menyimpan data User \n" + error.Message + "\nHubungi Developer segera !");
28. }} })

Pada segmen program 5.2, baris 6 - 15 terjadi pengecekan terhadap field yang diisi agar panjang tidak melebihi batas. Pada baris 17 query untuk insert user disusun. Pada baris 21-22 query dieksekusi. Apabila eksekusi berhasil maka akan muncul *messagebox* bahwa berhasil menambah user. Apabila eksekusi query gagal akan masuk ke baris 25-27 dan muncul *messagebox* bahwa terjadi kesalahan saat menambah user.

## 5.3 Tambah Ke Keranjang

Pada bagian ini Pemilik toko atau akuntan dapat menambahkan ke Keranjang barang - barang yang ingin dibeli oleh para customer, pemilik toko dapat memilih barang dan memasukan ke dalam cart, di dalam cart bisa di edit harga barang sesuai nego oleh pemilik toko.

**Segmen Program 5.3 Tambah Ke Keranjang**

1. private void add\_to\_cart(DataGridViewCellEventArgs e){
2. int index = e.RowIndex
3. if (index != -1){
4. int columnStok = -1;
5. if (whoLogin == "akuntan"){
6. columnStok = 4;}
7. else if (whoLogin == "kasir"){
8. columnStok = 3;}
9. if (int.Parse(dgvBarang.Rows[index].Cells[columnStok].Value.ToString()) > 0){
10. FormInputJumlahBeli formInputJumlahBeli = new FormInputJumlahBeli();
11. formInputJumlahBeli.setBatasBeli(dgvBarang.Rows[index].Cells[columnStok].Value.ToString(), this);
12. formInputJumlahBeli.ShowDialog();
13. if (jumlahBeli != -1){
14. int ctr = -1; // -1 = tidak ada
15. for (int i = 0; i < dgvCart.Rows.Count; i++){
16. if (dgvBarang.Rows[index].Cells[0].Value.ToString() == dgvCart.Rows[i].Cells[0].Value.ToString()){
17. ctr = i;}}
18. try{
19. if (ctr == -1) {
20. DataRow dr = tableCart.NewRow();
21. dr[0] = dgvBarang.Rows[index].Cells[0].Value.ToString(); // ID Barang
22. dr[1] = dgvBarang.Rows[index].Cells[1].Value.ToString(); // Nama Barang
23. dr[2] = jumlahBeli; // Jumlah Beli
24. if (whoLogin == "akuntan"){
25. dr[3] = dgvBarang.Rows[index].Cells[5].Value.ToString(); ;
26. //dr[4] = dgvBarang.Rows[index].Cells[2].Value.ToString();
27. //dr[5] = Convert.ToInt32(dr[2]) \* Convert.ToInt32(dr[4]); // Subtotal // tidak dipakai sejak 28 September 2022
28. dr[6] = dgvBarang.Rows[index].Cells[8].Value.ToString(); // Jenis barang}
29. else if (whoLogin == "kasir"){
30. dr[3] = dgvBarang.Rows[index].Cells[4].Value.ToString(); ;
31. //dr[4] = dgvBarang.Rows[index].Cells[2].Value.ToString(); // Harga jual // tidak dipakai sejak 28 September 2022
32. //dr[5] = Convert.ToInt32(dr[2]) \* Convert.ToInt32(dr[4]); // Subtotal // tidak dipakai sejak 28 September 2022
33. dr[6] = dgvBarang.Rows[index].Cells[7].Value.ToString(); // Jenis barang}
34. tableCart.Rows.Add(dr);
35. dgvCart.DataSource = tableCart;}
36. else{
37. if ((int.Parse(dgvCart.Rows[ctr].Cells[2].Value.ToString()) + jumlahBeli) <= int.Parse(dgvBarang.Rows[index].Cells[4].Value.ToString()))tableCart.Rows[ctr].SetField(2, (int.Parse(dgvCart.Rows[ctr].Cells[2].Value.ToString()) + jumlahBeli));
38. if (dgvCart.Rows[ctr].Cells[4].Value != null && dgvCart.Rows[ctr].Cells[4].Value.ToString() != "")}
39. tableCart.Rows[ctr].SetField(5, (int.Parse(dgvCart.Rows[ctr].Cells[2].Value.ToString()) \* int.Parse(dgvCart.Rows[ctr].Cells[4].Value.ToString())));}
40. dgvCart.DataSource = tableCart;}
41. else{
42. MessageBox.Show("Input jumlah beli melampaui stok!");} }
43. }catch (Exception error){
44. MessageBox.Show("Error saat memasukkan barang ke cart \n" + error.Message + "\nHubungi Developer segera !");}
45. jumlahBeli = -1;
46. count\_total\_belanja();}
47. else {
48. MessageBox.Show("Input jumlah beli dahulu!");}
49. }else{
50. MessageBox.Show("Stok Habis!");
51. }} }

Pada segmen program 5.3, baris 20 - 35 adalah kode untuk menambahkan barang yang belum ada di keranjang ke tabel keranjang bernama tableCart yang terhubung dengan *datagridview* keranjang. Pada baris 37 - 42 berfungsi untuk menambahkan barang yang akan dibeli dan sudah ada di keranjang, kemudian berubah adalah jumlah beli. Juga ada pengecekan apabila jumlah barang yang dibeli melebihi stok yaitu pada baris 38.

## 5.4 Checkout Htrans

Pada bagian ini Pemilik toko atau akuntan dapat menambahkan ke Keranjang barang - barang yang ingin dibeli oleh para customer, pemilik toko dapat memilih barang dan memasukan ke dalam cart, di dalam cart bisa di edit harga barang sesuai nego oleh pemilik toko.

**Segmen Program 5.4 Checkout Htrans**

1. string autogenhtrans = "select autogenhtrans from dual";
2. OracleCommand orclgen = new OracleCommand(autogenhtrans, conn);
3. String idhtrans = orclgen.ExecuteScalar().ToString();
4. //mendapatkan tanggal
5. string datenow = DateTime.Now.ToString("dd/MMM/yyyy");
6. // mendapatkan jam
7. string timenow = DateTime.Now.ToString("HH:mm:ss");
8. int isPerluKirim = -1; // -1 tidak dikirim, 0 perlu dikirim
9. for (int i = 0; i < dgvCart.Rows.Count; i++){
10. if (dgvCart.Rows[i].Cells[7].Value.ToString() == "Ya, dikirim")
11. {isPerluKirim = 0; }}
12. string queryInsHtrans = "";
13. if (isCash == 0 && !cbUangMuka.Checked) //transfer dan tidak dp
14. {queryInsHtrans = "insert into htrans (htrans\_id, htrans\_pembeli, " +
15. "htrans\_alamat, htrans\_tanggal," +" htrans\_jam, htrans\_total, htrans\_is\_cash, htrans\_jumlah\_bayar, htrans\_is\_terkirim\_semua, htrans\_penjual, htrans\_is\_lunas) " +
16. "values('" + idhtrans + "', '" + tableCustomer.Rows[cmbIDPembeli.SelectedIndex][0].ToString() + "', " +
17. "'" + txtAlamatPembeli.Text + "', TO\_DATE('" + datenow + "', 'dd/mm/yyyy'), " +
18. "TO\_DATE('" + timenow + "', 'hh24:mi:ss'), " + total + ", " + isCash + ", 0, " + isPerluKirim + ", '" + username + "', 0)";
19. }else if (isCash == 0 && cbUangMuka.Checked) //metode transfer dan tidak dp {
20. queryInsHtrans = "insert into htrans (htrans\_id, htrans\_pembeli, " +"htrans\_alamat, htrans\_tanggal," +
21. " htrans\_jam, htrans\_total, htrans\_is\_cash, htrans\_jumlah\_bayar, htrans\_is\_terkirim\_semua, htrans\_penjual, htrans\_is\_lunas) " +"values('" + idhtrans + "', '" + tableCustomer.Rows[cmbIDPembeli.SelectedIndex][0].ToString() + "', " +
22. "'" + txtAlamatPembeli.Text + "', TO\_DATE('" + datenow + "', 'dd/mm/yyyy'), " +
23. "TO\_DATE('" + timenow + "', 'hh24:mi:ss'), " + total + ", " + isCash + ", " + txtUangMuka.Text + ", " + isPerluKirim + ", '" + username + "', 0)";}
24. if (isCash == 1 && !cbUangMuka.Checked) //cash dan tidak dp
25. {queryInsHtrans = "insert into htrans (htrans\_id, htrans\_pembeli, " +"htrans\_alamat, htrans\_tanggal, " +
26. "htrans\_jam, htrans\_total, htrans\_is\_cash, htrans\_is\_terkirim\_semua, htrans\_penjual, htrans\_is\_lunas) " +
27. "values('" + idhtrans + "', '" + tableCustomer.Rows[cmbIDPembeli.SelectedIndex][0].ToString() + "'," +
28. '" + txtAlamatPembeli.Text + "', TO\_DATE('" + datenow + "', 'dd/mm/yyyy'), " +
29. "TO\_DATE('" + timenow + "', 'hh24:mi:ss')," + total + ", " + isCash + ", " + isPerluKirim + ", '" + username + "', 1)";
30. }else if (isCash == 1 && cbUangMuka.Checked) // metode cash dan dp{
31. queryInsHtrans = "insert into htrans (htrans\_id, htrans\_pembeli, " +"htrans\_alamat, htrans\_tanggal, " +
32. "htrans\_jam, htrans\_total, htrans\_is\_cash, htrans\_jumlah\_bayar, htrans\_is\_terkirim\_semua, htrans\_penjual, htrans\_is\_lunas) " + "values('" + idhtrans + "', '" + tableCustomer.Rows[cmbIDPembeli.SelectedIndex][0].ToString() + "'," +
33. " '" + txtAlamatPembeli.Text + "', TO\_DATE('" + datenow + "', 'dd/mm/yyyy'), " +
34. "TO\_DATE('" + timenow + "', 'hh24:mi:ss')," + total + ", " + isCash + ", " + txtUangMuka.Text + ", " + isPerluKirim + ", '" + username + "', 0)";}
35. OracleCommand insheader = new OracleCommand(queryInsHtrans, conn);
36. insheader.ExecuteNonQuery();

Pada segmen program 5.4, baris 1 - 3 adalah code untuk mengambil generate id htrans. Pada baris 14 - 34, adalah proses menyusun query insert htrans menyesuaikan dengan apakah ada uang muka dan metode pembayaran yang dipakai. Pada baris 35 - 36, adalah proses eksekusi query untuk menyimpan.

## 5.5 Checkout Dtrans

Pada bagian ini akan dijelaskan kegunaan dari Dtrans, Dtrans ini berguna untuk code generate dari dtrans. Tiap ada barang yang dibeli program akan meng generate code yang akan ditampilkan di *datagridview*.

**Segmen Program 5.5 Checkout Dtrans**

1. for (int i = 0; i < dgvCart.Rows.Count; i++)
2. {
3. get autogen dtrans
4. string autogendtrans = "select autogendtrans from dual";
5. OracleCommand orcldgen = new OracleCommand(autogendtrans, conn);
6. String iddtrans = orcldgen.ExecuteScalar().ToString();
7. string queryGetHargaKulakan = "select barang\_harga\_kulakan from barang where barang\_id = '" + dgvCart.Rows[i].Cells[0].Value.ToString() + "'";
8. OracleCommand oracleCommandHargaKulakan = new OracleCommand(queryGetHargaKulakan, conn);
9. String hargaKulakan = oracleCommandHargaKulakan.ExecuteScalar().ToString();
10. string queryInsDtrans = "insert into dtrans values ('" + iddtrans + "', '" + idhtrans + "', '" + dgvCart.Rows[i].Cells[0].Value.ToString() + "', "
11. + dgvCart.Rows[i].Cells[2].Value.ToString() + ", '" + dgvCart.Rows[i].Cells[3].Value.ToString() + "', "
12. + formatHarga.ubahFormatDariCurrencyKeInt(dgvCart.Rows[i].Cells[5].Value.ToString()) + ", " + hargaKulakan + "";
13. if (dgvCart.Rows[i].Cells[7].Value.ToString() == "Ya, dikirim"){queryInsDtrans += ", 0)";}else
14. {queryInsDtrans += ", null)"; }
15. OracleCommand insdetail = new OracleCommand(queryInsDtrans, conn);
16. insdetail.ExecuteNonQuery();
17. string queryGetStokBarang = "select barang\_stok from barang where barang\_id = '" + dgvCart.Rows[i].Cells[0].Value.ToString() + "'";
18. OracleCommand oracleCommandStokBarang = new OracleCommand(queryGetStokBarang, conn);
19. String tempStokBarang = oracleCommandStokBarang.ExecuteScalar().ToString();
20. int temp\_stok = int.Parse(tempStokBarang) - int.Parse(dgvCart.Rows[i].Cells[2].Value.ToString());
21. string queryUpdBarang = "update barang set barang\_stok = " + temp\_stok + " where barang\_id = '" + dgvCart.Rows[i].Cells[0].Value.ToString() + "'";
22. OracleCommand updStokBarang = new OracleCommand(queryUpdBarang, conn);
23. updStokBarang.ExecuteNonQuery();
24. }

Pada segmen program 5.5, baris 4 - 6 adalah code untuk mengambil generate id dtrans. Pada baris 7 - 9 adalah code untuk mengambil data harga kulak dari barang tersebut. Pada baris 10 - 16 proses penyusunan query dan eksekusi untuk menyimpan data barang yang dibeli. Pada baris 17 - 23 adalah query dan code untuk mengurangi stok dari barang yang dibeli. Hal ini terus berulang hingga seluruh barang tersimpan dengan baik.

## 5.6 Tambah Barang

Pada bagian ini Pemilik toko dapat menambahkan barang yang ingin dijual atau dimasukan ke stock pada halaman ini. Setelah itu barang akan masuk kedalam stok barang atau catalog barang, yang bisa menambahkan barang hanya role akuntan.

**Segmen Program 5.6 Tambah Barang**

1. if (txtNamaBarang.Text != "" && (nudHargaJualBarang.Value > 0) && (nudStockBarang.Value > 0) && cmbGudang.SelectedIndex > -1
2. && txtSuplier.Text != "" && (nudHargaBeliBarang.Value > 0) && txtSatuan.Text != "" && cmbKategori.SelectedIndex > -1)
3. {if (txtNamaBarang.Text.Length > 100)
4. { MessageBox.Show("Nama Barang tidak boleh lebih dari 100 karakter");
5. return;}
6. if (txtSuplier.Text.Length > 100)
7. {MessageBox.Show("Nama Supplier tidak boleh lebih dari 100 karakter");
8. return;
9. }if (txtSatuan.Text.Length > 10)
10. {MessageBox.Show("Satuan Barang tidak boleh lebih dari 10 karakter");
11. return;}
12. try{
13. if (conn.State == ConnectionState.Closed)
14. {conn = new OracleConnection(connStr);
15. conn.Open();}
16. oracleTransaction = conn.BeginTransaction();
17. String idbarang = "";
18. if (conn.State == ConnectionState.Closed)
19. {conn = new OracleConnection(connStr);
20. conn.Open();}
21. string queryInsertBarang = "insert into barang (barang\_id, barang\_nama, barang\_harga\_jual, barang\_harga\_kulakan, barang\_stok, barang\_satuan, barang\_id\_gudang, barang\_id\_kategori, barang\_is\_deleted) values " +
22. "('" + idbarang + "', '" + txtNamaBarang.Text + "', " + nudHargaJualBarang.Value + ", " + nudHargaBeliBarang.Value + ", " + nudStockBarang.Value +
23. ", '" + txtSatuan.Text + "', '" + tableGudang.Rows[cmbGudang.SelectedIndex]["GUDANG\_ID"].ToString() + "', '" + tableKategori.Rows[cmbKategori.SelectedIndex]["KATEGORI\_ID"].ToString() + "'" +
24. ", 0)";
25. // insert data barang
26. OracleCommand oracleCommand = new OracleCommand(queryInsertBarang, conn);
27. oracleCommand.ExecuteNonQuery();
28. get autogen id pembeliansupplier
29. string idpembeliansupplier = "";
30. string autogenpembeliansupplier = "select autogenpembeliansupplier from dual";
31. orclgen = new OracleCommand(autogenpembeliansupplier, conn);
32. idpembeliansupplier = orclgen.ExecuteScalar().ToString();
33. /mendapatkan tanggal
34. string getdate = dateTanggal.Value.ToString("dd/MMM/yyyy");
35. insert pembeliansupplier
36. string queryInsertPembelianSupplier = "insert into pembeliansupplier values " +
37. "('" + idpembeliansupplier + "', '" + idbarang + "', " + nudStockBarang.Value + ", '" + txtSatuan.Text + "'" +
38. ", '" + txtSuplier.Text + "', TO\_DATE('" + getdate + "', 'dd/mm/yyyy'))";
39. oracleCommand = new OracleCommand(queryInsertPembelianSupplier, conn);
40. oracleCommand.ExecuteNonQuery();
41. oracleTransaction.Commit();
42. MessageBox.Show("Berhasil simpan data barang " + txtNamaBarang.Text + " !");
43. resetAll();}
44. catch (Exception error)
45. {
46. oracleTransaction.Rollback();
47. MessageBox.Show("Error saat insert barang \n" + error.Message + "\nHubungi Developer segera !");}
48. }else{
49. MessageBox.Show("Pastikan semua field terisi!");
50. }

Pada segmen program 5.6, baris 3 - 11 adalah pengecekan batasan panjang dari data yang ingin disimpan. Pada baris 21 - 24 adalah penyusunan query untuk menyimpan barang baru ke database. Eksekusi query insert barang baru ke database dilakukan oleh code pada baris 26 - 27. Tambah barang baru ini dianggap sebagai pembelian barang baru dari supplier maka dari itu akan disimpan pada tabel pembeliansupplier melalui code pada baris 34 - 40. Proses pengambilan hasil generate id pembeliansupplier ada pada baris 29 - 32.

## 5.7 Edit Barang

Pada bagian ini Pemilik toko dapat mengedit barang yang ada. Edit barang dapat memperbarui harga, stock ataupun nama barang. Barang yang di edit akan otomatis terganti di katalog dengan hasil yang telah di edit.

**Segmen Program 5.7 Edit Barang**

1. Private void btnEdit\_Click(object sender, EventArgs e)
2. { DialogResult dialogResult;
3. dialogResult = MessageBox.Show("Apakah Ingin Update data Barang ?", "Warning", MessageBoxButtons.YesNo, MessageBoxIcon.Information);
4. if (dialogResult == DialogResult.Yes)
5. {if (txtNamaBarang.Text != "" && (nudHargaJualBarang.Value > 0) && (nudStockBarang.Value > 0) && cmbGudang.SelectedIndex > -1
6. && txtSuplier.Text != "" && (nudHargaBeliBarang.Value > 0) && txtSatuan.Text != "" && cmbKategori.SelectedIndex > -1)
7. {if (txtNamaBarang.Text.Length > 100){
8. MessageBox.Show("Nama Barang tidak boleh lebih dari 100 karakter");
9. return; }
10. if (txtSuplier.Text.Length > 100)
11. {MessageBox.Show("Nama Supplier tidak boleh lebih dari 100 karakter");
12. return;
13. if (txtSatuan.Text.Length > 10)
14. {MessageBox.Show("Satuan Barang tidak boleh lebih dari 10 karakter");
15. return;
16. string datenow = DateTime.Now.ToString("dd/MMM/yyyy");
17. string queryUpdBarang = "update barang set barang\_nama = '" + txtNamaBarang.Text + "', barang\_harga\_jual = " + nudHargaJualBarang.Value + ", barang\_harga\_kulakan = " + nudHargaBeliBarang.Value + ", barang\_stok = " + nudStockBarang.Value +
18. ", barang\_satuan = '" + txtSatuan.Text + "', barang\_id\_gudang = '" + tableGudang.Rows[cmbGudang.SelectedIndex]["GUDANG\_ID"].ToString() + "', barang\_id\_kategori = '" + tableKategori.Rows[cmbKategori.SelectedIndex]["KATEGORI\_ID"].ToString() + "'" +
19. ", barang\_tgl\_edit = TO\_DATE('" + datenow + "', 'dd/mm/yyyy')" +
20. " where barang\_id = '" + lblIDBarang.Text + "'";
21. try{if (conn.State == ConnectionState.Closed)
22. {conn = new OracleConnection(connStr);
23. conn.Open();}
24. oracleTransaction = conn.BeginTransaction();
25. OracleCommand oracleCommand = new OracleCommand(queryUpdBarang, conn);
26. oracleCommand.ExecuteNonQuery();
27. if (nudStockBarang.Value > int.Parse(nudStockBarang.Tag.ToString()))
28. { // get autogen id pembeliansupplier
29. string idpembeliansupplier = "";
30. string autogenpembeliansupplier = "select autogenpembeliansupplier from dual";
31. OracleCommand orclgen = new OracleCommand(autogenpembeliansupplier, conn);
32. idpembeliansupplier = orclgen.ExecuteScalar().ToString() //mendapatkan tanggal
33. tring getdate = dateTanggal.Value.ToString("dd/MMM/yyyy");
34. / insert pembeliansupplier
35. string queryInsertPembelianSupplier = "insert into pembeliansupplier values " +
36. "('" + idpembeliansupplier + "', '" + lblIDBarang.Text + "', " + (nudStockBarang.Value - int.Parse(nudStockBarang.Tag.ToString())) + ", '" + txtSatuan.Text + "'" +
37. ", '" + txtSuplier.Text + "', TO\_DATE('" + getdate + "', 'dd/mm/yyyy'))";
38. oracleCommand = new OracleCommand(queryInsertPembelianSupplier, conn);
39. oracleCommand.ExecuteNonQuery();
40. }
41. oracleTransaction.Commit();
42. MessageBox.Show("Berhasil ubah data Barang " + lblIDBarang.Text + "!");
43. resetAll();}
44. catch (Exception error)
45. {oracleTransaction.Rollback();
46. MessageBox.Show("Error saat ubah data Barang \n" + error.Message + "\nHubungi Developer segera !");
47. }}}}

Pada segmen program 5.7, baris 7 - 15 terdapat pengecekan untuk panjang dari inputan agar tidak terjadi error ketika melakukan pengubahan data barang. Pada baris 35 - 37, adalah penyusunan query untuk update ke tabel kategori di database. pada baris 38 - 39 adalah code untuk eksekusi query update ke tabel barang. Apabila eksekusi berhasil maka akan muncul pesan berhasil dari *messagebox*. Apabila eksekusi gagal maka akan muncul pesan gagal dari *messagebox* sekaligus dengan pesan errornya.

## 5.8 Tambah Kategori

Pada bagian ini Pemilik toko dapat mengedit barang yang ada. Edit barang dapat memperbarui harga, stock ataupun nama barang. Barang yang di edit akan otomatis terganti di katalog dengan hasil yang telah di edit.

**Segmen Program 5.8 Tambah Kategori**

1. private void btnTambah\_Click(object sender, EventArgs e)
2. {DialogResult dialogResult;
3. dialogResult = MessageBox.Show("Apakah Anda yakin ingin menambah data kategori ?", "Warning", MessageBoxButtons.YesNo, MessageBoxIcon.Information);
4. if (dialogResult == DialogResult.Yes)
5. {if (txtNamaKategori.Text != "")
6. {if (txtNamaKategori.Text.Length > 100)
7. {MessageBox.Show("Nama Kategori tidak boleh lebih dari 100 karakter");
8. return; }
9. String idkategori = "";
10. try{
11. if (conn.State == ConnectionState.Closed)
12. {conn = new OracleConnection(connStr);
13. conn.Open();}
14. // get autogen hutangsupplier
15. string autogenkategori = "select autogenkategori from dual";
16. oracleCommand orclgen = new OracleCommand(autogenkategori, conn);
17. idkategori = orclgen.ExecuteScalar().ToString();
18. }atch (Exception error)
19. {MessageBox.Show("Error saat ambil id kategori \n" + error.Message + "\nHubungi Developer segera !");
20. return;
21. string queryInsKategori = "insert into kategori values ('" + idkategori + "', '" + txtNamaKategori.Text + "', 0)";
22. try
23. OracleCommand oracleCommand = new OracleCommand(queryInsKategori, conn);
24. oracleCommand.ExecuteNonQuery();
25. MessageBox.Show("Berhasil insert data kategori " + txtNamaKategori.Text + "!");
26. resetAll();}
27. catch (Exception error)
28. { MessageBox.Show("Error saat insert kategori \n" + error.Message + "\nHubungi Developer segera !");
29. }}
30. else{MessageBox.Show("Pastikan field nama kategori terisi!");
31. }}}

Pada segmen program 5.8, baris 6 - 8 terdapat pengecekan untuk panjang dari inputan agar tidak terjadi error ketika melakukan penambahan data kategori. Pada baris 15 - 17 adalah code untuk memanggil *function* autogenkategori untuk mengambil hasil generate id kategori baru. Pada baris 21, adalah penyusunan query untuk insert ke tabel kategori di database. pada baris 23 - 24 adalah code untuk eksekusi query insert ke tabel kategori. Apabila eksekusi berhasil maka akan muncul pesan berhasil dari *messagebox*. Apabila eksekusi gagal maka akan muncul pesan gagal dari *messagebox* sekaligus dengan pesan errornya.

## 5.9 Edit Kategori

Pada bagian ini Pemilik toko dapat mengedit barang yang ada. Edit barang dapat memperbarui harga, stock ataupun nama barang. Barang yang di edit akan otomatis terganti di catalog dengan hasil yang telah di edit.

**Segmen Program 5.9 Edit Kategori**

1. private void btnEdit\_Click(object sender, EventArgs e)
2. {DialogResult dialogResult;
3. dialogResult = MessageBox.Show("Apakah Anda yakin ingin simpan ubahan nama kategori ?", "Warning", MessageBoxButtons.YesNo, MessageBoxIcon.Information);
4. if (dialogResult == DialogResult.Yes)
5. {if (txtNamaKategori.Text != "")
6. {if (txtNamaKategori.Text.Length > 100)
7. {MessageBox.Show("Nama Kategori tidak boleh lebih dari 100 karakter");
8. return; }
9. string queryEditKategori = "update kategori set kategori\_nama = '" + txtNamaKategori.Text + "' where kategori\_id = '" + txtIDKategori.Text + "'";
10. try{if (conn.State == ConnectionState.Closed)
11. {conn = new OracleConnection(connStr);
12. conn.Open();}
13. OracleCommand oracleCommand = new OracleCommand(queryEditKategori, conn);
14. oracleCommand.ExecuteNonQuery();
15. MessageBox.Show("Berhasil update data kategori " + txtIDKategori.Text + "!");
16. resetAll();}
17. catch (Exception error)
18. {
19. MessageBox.Show("Error saat update data Kategori \n" + error.Message + "\nHubungi Developer segera !");
20. } }
21. else{
22. MessageBox.Show("Pastikan field nama kategori tidak kosong !");
23. }}

Pada segmen program 5.9, pada baris 5 - 8 terdapat pengecekan untuk panjang dari inputan agar tidak terjadi error ketika melakukan edit. Pada baris 9 terjadi penyusunan query. Pada bari 13 - 14, query yang sudah tersusun dieksekusi. Apabila eksekusi berhasil maka akan muncul *messagebox* berisi pesan bahwa berhasil melakukan perubahan dengan code seperti pada baris 15. Apabila terjadi kegagalan maka akan muncul *messagebox* berisi pesan bahwa gagal melakukan perubahan sekaligus pesan error dengan code seperti pada baris 19.

## 5.10 Autogendtrans

Pada bagian ini adalah sebuah *function* yang berfungsi untuk membuat auto generate dari dtrans. Auto generate ini berfungsi untuk generate id dtrans. Code ini disimpan pada Oracle. Ketika diperlukan maka akan dipanggil oleh program. Function ini akan mengembalikan hasil generate id dtrans.

**Segmen Program 5.10 Autogendtrans**

1. create or replace function autogendtrans
2. return varchar2
3. is
4. temp varchar(20);
5. begin
6. select count(DTRANS\_ID) into temp from DTRANS;
7. return 'DT' || lpad(to\_number(temp)+1,18,'0');
8. end;
9. /
10. show err;

Pada segmen program 5.10, baris 6 digunakan untuk menghitung berapa banyak data dtrans yang tersimpan kemudian dijadikan id dtrans baru. Penggabungan tersebut terjadi pada baris 7 menjadi satu kesatuan utuh. Pada baris 7 juga akan memberikan hasil ke pemanggil *function*.

## 5.11 Autogenhtrans

Pada bagian ini adalah sebuah *function* yang berfungsi untuk membuat auto generate id dari tabel htrans. Auto generate ini berfungsi untuk generate id htrans. Code ini disimpan pada Oracle. Ketika diperlukan maka akan dipanggil oleh program. Function ini akan mengembalikan hasil generate id htrans.

**Segmen Program 5.11 Autogenhtrans**

1. create or replace function autogenhtrans
2. return varchar2
3. is
4. temp varchar(20);
5. begin
6. select count(HTRANS\_ID) into temp from HTRANS;
7. return 'HT' || lpad(to\_number(temp)+1,18,'0');
8. end;
9. /
10. show err;

Pada segmen program 5.11, baris 6 digunakan untuk menghitung berapa banyak data pada tabel htrans yang tersimpan di database kemudian dijadikan id htrans baru. Penggabungan tersebut terjadi pada baris 7 menjadi satu kesatuan utuh. Pada baris 7 juga akan memberikan hasil ke pemanggil *function*.

## 5.12 Autogenhutangsupplier

Pada bagian ini adalah sebuah *function* yang berfungsi untuk membuat auto generate id dari tabel pembeliansupplier. Auto generate ini berfungsi untuk generate id pembeliansupplier. Code ini disimpan pada Oracle. Ketika diperlukan maka akan dipanggil oleh program. Function ini akan mengembalikan hasil generate id pembeliansupplier.

**Segmen Program 5.12 Autogenhtrans**

1. create or replace function autogenhutangsupplier
2. return varchar2
3. is
4. temp varchar(20);
5. begin
6. select count(hutangsupplier\_ID) into temp from hutangsupplier;
7. return 'HS' || lpad(to\_number(temp)+1,18,'0');
8. end;
9. /
10. show err;

Pada segmen program 5.12, baris 6 digunakan untuk menghitung berapa banyak data pada tabel pembeliansupplier yang tersimpan di database kemudian dijadikan id pembeliansupplier baru. Penggabungan tersebut terjadi pada baris 7 menjadi satu kesatuan utuh. Pada baris 7 juga akan memberikan hasil ke pemanggil *function*.

## 5.13 Autogensuratjalan

Pada bagian ini adalah sebuah *function* yang berfungsi untuk membuat auto generate id dari tabel suratjalan. Auto generate ini berfungsi untuk generate id suratjalan. Code ini disimpan pada Oracle. Ketika diperlukan maka akan dipanggil oleh program. Function ini akan mengembalikan hasil generate id suratjalan.

**Segmen Program 5.13 Autogensuratjalan**

1. create or replace function autogensuratjalan
2. return varchar2
3. is
4. temp varchar(20);
5. begin
6. select count(SURATJALAN\_ID) into temp from SURATJALAN;
7. return 'SJ' || lpad(to\_number(temp)+1,18,'0');
8. end;
9. /
10. show err;

Pada segmen program 5.13, baris 6 digunakan untuk menghitung berapa banyak data pada tabel suratjalan yang tersimpan di database kemudian dijadikan id suratjalan baru. Penggabungan tersebut terjadi pada baris 7 menjadi satu kesatuan utuh. Pada baris 7 juga akan memberikan hasil ke pemanggil *function*.